AMENDMENTS TO THE CLAIMS
(with complete listing)

1. (Cancelled)

2. (Currently amended) The method of claim [[1]]28 further comprising the steps of[[,]]:

writing ~~including~~ a condition statement (3147) in said at least one template, said condition statement having a condition expression contained therein,

evaluating said condition expression,

performing a first task if said condition expression is evaluated to be a first value, and

performing a second task if said condition expression is evaluated to be a second value.

3. (Currently amended) The method of claim [[1]]28 further comprising the steps of[[,]]:

writing ~~including~~ an iteration statement (3148) in said at least one template, said iteration statement having an iteration expression contained therein,

iteratively evaluating said iteration expression, and

performing a task while said iteration expression is evaluated to be a particular value.

4. (Currently amended) The method of claim [[1]]28 further comprising the steps of[[,]]:

writing ~~including~~ a subroutine statement (3149) in said at least one template, said subroutine statement having a subroutine name contained therein,

designating a portion of said at least one template by said subroutine name, and

interpreting said subroutine statement as a directive to control said writing by said portion.

5. (Cancelled)

6. (Currently amended) The method of claim [[1]]28 further comprising the steps of[[,]]:

writing ~~including~~ a set statement in said at least one template, said set statement having a set expression and a set variable contained therein,

assigning said set expression to said set variable.

7. (Cancelled)

8. (Currently amended) The system of claim [[7]]29 wherein,

said deconstruction program includes a language-determination parser (206).

9. (Currently amended) The system of claim [[7]]29 wherein,

said deconstruction program includes at least one language-dependent parser (208).

10. (Cancelled)

11. (Currently amended) The system of claim ~~7 wherein,~~ 29 further comprising:

~~said database stores~~ an ~~original~~ environment description of said source database system stored in said relational common construct database.

12. (Currently amended) The system of claim ~~7 wherein,~~ 29 further comprising:

~~said database stores~~ a target environment description stored in said relational common construct database.

13–15. (Cancelled)

16. (Currently amended) The system of claim [[7]]29 wherein,

said construction program includes a template workbench (410) designed and arranged for editing said ~~at least one~~ template.

17. (Cancelled)

18. (Currently amended) The system of claim [[7]]29 wherein,

said algorithmic ~~template~~ language includes a condition statement (3147) having a condition expression contained therein,

said construction program being designed and arranged to interpret said condition statement as a directive to evaluate said condition expression and perform a first task if said

condition expression is evaluated to be a first value and perform a second task if said condition expression is evaluated to be a second value.

19. (Currently amended) The system of claim [[7]]29 wherein,

said algorithmic ~~template~~ language includes an iteration statement (3148) having an iteration expression contained therein,

said construction program being designed and arranged to interpret said iteration statement as a directive to iteratively evaluate said iteration expression and perform a task while said iteration expression is evaluated to be a particular value.

20. (Currently amended) The system of claim [[7]]29 wherein,

said algorithmic ~~template~~ language includes a subroutine statement (3149) having a subroutine name contained therein,

said ~~at least one~~ template (314) having a portion therein designated by said subroutine name,

said construction program being designed and arranged to interpret said subroutine statement as a directive to interpret said portion,

wherein when said construction program has completed interpreting said portion, said construction program interprets said algorithmic language immediately following said subroutine statement.

21. (Cancelled)

22. (Currently amended) The system of claim [[7]]29 wherein,

said algorithmic ~~template~~ language includes a set statement having a set expression and a set variable contained therein,

said construction program being designed and arranged to interpret said set statement as a directive to assign said set expression to said set variable.

23. (Currently amended) The [[A]] system of claim 29 wherein, ~~for converting original computer source code (100) to target source code (500) comprising,~~

~~at least one computer system having a deconstruction means (20), a database (30), at least one template (314), and a construction means (40),~~

~~said deconstruction means being arranged and designed to resolve said original source code into a plurality of basic constituent elements (3000) and a plurality of interrelationships therebetween,~~

~~said database adapted for storing said elements and interrelationships,~~

said template ~~defining~~ generally defines the structure of a ~~said~~ target computer program source code file of said target database system, and ~~comprising an algorithmic template language (3144) designed and arranged for controlling said construction means, wherein~~

said construction program ~~means~~ is arranged and designed to interpret said template and write said target computer program source code file therefrom, ~~incorporating said elements and interrelationships as directed by said template~~.

24. (Currently amended) The system ~~method~~ of claim 23 wherein,

said algorithmic ~~template~~ language includes a condition evaluation means to control said writing in a first manner if said condition expression is evaluated to be a first value and control said writing in a second manner if said condition expression is evaluated to be a second value.

25. (Currently amended) The system ~~method~~ of claim 23 wherein,

said algorithmic ~~template~~ language includes a looping means to iteratively evaluate a looping expression and iteratively control said writing in a predetermined manner while said iteration expression is evaluated to be a particular value.

26. (Currently amended) The system ~~method~~ of claim 23 wherein,

said algorithmic ~~template~~ language includes a means to include subroutines.

27. (Currently amended) The system ~~method~~ of claim 23 wherein,

said template language includes a means to assign a variable a predetermined value.

28. (New) A method for converting a source legacy database system (10) to a target relational database system (40), the source legacy database system defining an environment that includes a first non-source-code data file and a first computer program source code file (110), the method comprising the steps of:

collecting the entirety (302) of said source legacy database;

storing said entirety in a relational common construct database (30);

resolving said entirety into a plurality of basic constituent elements (3000) thereof;

storing said plurality of basic constituent elements in said relational common construct database;

writing at least one template (314) in an algorithmic language;

storing said at least one template in said relational common construct database; and

writing said target relational database from said at least one template as a function of said plurality of basic constituent elements, said at least one template controlling said writing.

29. (New) A system for converting a source database system (302) to a target database system (318), said source database system composed of a plurality of basic constituents (3000) that are arranged to form at least a first non-source-code data file and a first computer

program source code file (110), said target database having substantially equivalent functionality as said source database, the system comprising:

at least one computer system;

a relational database management system structured for execution by said at least one computer system;

a deconstruction program (20) structured for execution by said at least one computer system;

a construction program (40) structured for execution by said at least one computer system;

a relational common construct database (30) operably coupled by said relational database management system to said deconstruction program and said construction program;

said source database system stored in said relational common construct database;

said deconstruction program operatively coupled to said source database system for resolving said source database system into said plurality of basic constituents;

each of said plurality of basic constituents individually stored in said relational common construct database;

a template (314) written in an algorithmic language stored in said relational common construct database and operatively coupled to said construction program for controlling said construction program in writing said target database system as a function of said plurality of basic constituents,

said construction program operatively coupled to said target database system; and

said target database system stored in said relational common construct database.

30. (New) A system for converting a source database system (302) to a target database system (318), said source database system defining an environment that includes a first non-source-code data file and a first software program source code file (110), the system comprising:

at least one computer system having a relational database management system and a common construct database (30);

said first data file and said first software program source code file stored in said common construct database by said relational database management system of said at least one computer system;

a conversion software program resident on said at least one computer system designed and arranged to receive as an input said first data file and said first software program source code file and to produce as an output said target database system;

said common construct database including a template written in an algorithmic language that controls said conversion software program and defines the structure of said target database system.

31. (New) The system of claim 30 wherein

said first data file is structured as a software report (102).

32. (New) The system of claim 30 wherein

said first data file is structured as a utility (104).

33. (New) The system of claim 30 wherein

said first data file is structured as a descriptor (106).

34. (New) The system of claim 30 wherein

said first data file is structured as a repository extract (108).

35. (New) The system of claim 30 wherein:

said source database system is a legacy database system; and

said target database system is a relational database system.

36. (New) The system of claim 29 wherein:

said source database system is a legacy database system; and

said target database system is a relational database system.